REMARKS / ARGUMENTS

Claims 1-30 are pending in the application.

Claims 1-4, 7-12, 14, 16-20, 25-27, and 29-30 are rejected under 35 U.S.C. §103(a) as

being unpatentable over U.S Patent No. 5,860,105 to McDermott (Hereinafter "McDermott") in

view of U.S Patent No. 6,490,657 to Masubuchi et al. (Hereinafter "Masubuchi"). Claims 5-6

are rejected under 35 U.S.C. §103(a) as being unpatentable over McDermott in view of

Masubuchi in further view of U.S Patent No. 6,058,456 to Arimilli (Hereinafter "Arimilli").

Claim 13 is rejected under 35 U.S.C. §103(a) as being unpatentable over McDermott in view of

Masubuchi in further view of U.S Patent No. 6,460,122 to Otterness et al. (Hereinafter

"Arimilli"). Claims 15 and 28 are rejected under 35 U.S.C. §103(a) as being unpatentable over

McDermott in view of Masubuchi in further view of U.S Patent No. 5,724,550 to Stevens

(Hereinafter "Stevens"). Claims 21-24 are rejected under 35 U.S.C. §103(a) as being

unpatentable over Masubuchi in view of McDermott.

Claim Rejections under 35 U.S.C. §103(a)

Claims 1-4, 7-12, 14, 16-20, 25-27, and 29-30 are rejected under 35 U.S.C. §103(a) as

being unpatentable over McDermott in view of Masubuchi. McDermott discloses an NDIRTY

cache line lookahead technique used to expedite cache flush and export operations by providing

a mechanism to avoid scanning at least some cache lines that do not contain dirty data (*See*

Abstract). Masabuchi discloses a cache flush apparatus, attached to the system bus, that

maintains the memory addresses held in dirty blocks within its own storage during the normal

data processing. When a cache flush is required, the cache flush apparatus reads the memory

addresses from the storage and issues bus commands. As a result, a dirty block becomes

"shared" and it still holds the same data. For a non-dirty cache block, it remains unchanged (*See* Col. 2, Lines 49-57).

Neither McDermott nor Masabuchi disclose a list structure for tracking a status of a plurality of cache entries, wherein the list structure is located outside a cache and does not contain cache data or addresses. The Office Action attempts to combine the two references to disclose this element. The Office Action states:

> Regarding Applicant's argument regarding the teachings of Masabuchi, it should be noted that Masabuchi is relied upon for teaching that which McDermott does not. McDermott does not teach the list structure located outside of a cache. Masabuchi is relied upon for teaching a list structure which is located outside of a cache. Hence, it is not clear why the Applicant is arguing that Masabuchi's list structure contains addresses, since Masabuchi was never relied upon for teaching that feature.

(Office Action, Page 12).

McDermott's list structure does not contain addresses because it is located inside the cache. Specifically, McDermott states:

> Referring to FIG. 2a, the NDIRTY cache line lookahead technique of the invention is implemented as part of the flush/export logic for the L1 cache 204.

(McDermott, Col. 6, Lines 57-59).

McDermott's list structure would be inoperable outside of the cache. Masabuchi does not make it possible for McDermott's list structure to operate outside of the cache. Masabuchi states:

> A cache flush device 30 and a memory controller 50 for a main memory 51, are connected to the system bus 40. The cache flush device 30 includes a system bus interface 31, an update address memory 32 (consisting of regions A_0 to A_{n-1}), an update address registering section 33, an update address removing section 34 and a flush executing section 35. The update address registering section 33 and the update address removing section 34 can be implemented as a single hardware module.
>
> The system bus interface 31 serves as an interface with the system bus 40.
>
> The update address memory 32 consists of n regions A_0 to A_{n-1} for storing addresses of

51928_1   -9-

> data held in the dirty blocks of the cache memories 20. In this embodiment, if and only if a dirty block holds data of a certain address, the address is stored in a certain region of the update address memory 32. From now on, we call such a memory address stored in the update address memory 32 an update address.

(Masabuchi, Col. 11, Lines 5-23).

The Masabuchi list structure is entirely a list of addresses. The addresses are necessitated by the external nature of the list structure. Combining Masabuchi and McDermott would not produce a workable external list structure that does not contain cache data or addresses. Therefore, a list structure for tracking a status of a plurality of cache entries, wherein the list structure is located outside a cache and does not contain cache data or addresses, as cited by claims 1, 16, and 21.

The Office Action further states:

> Additionally, it should be noted that a reference does not teach away from a particular modification merely because it is silent to making the modification. st because McDermott teaches a list structure within the cache does not necessarily infer that McDermott teaches away from having the list structure located outside of the cache.

(Office Action, Page 13).

McDermott does more than teach a list structure within the cache. McDermott states:

> An object of the invention is to store information in a cache array to reduce the time required for cache export and flush (export then invalidate) operations.

(McDermott, Col. 1, Lines 65-67).

McDermott teaches a list structure within the cache to speed up operation of the cache as an object of the invention. An external list would increase the time required for cache export and flush operations, which is counter to an express goal of the invention.

With respect to other references cited by Examiner, Arimilli, Otterness, and Stevens do not overcome the above noted deficiencies of McDermott and Masubuchi. In particular, Arimilli, Otterness, and Stevens do not teach or   suggest "a list structure for tracking a status

of a plurality of cache entries, wherein said list structure is located outside a cache and wherein said list structure does not contain cache data or addresses," as recited in independent claims 1 and 21, and currently amended independent claim 16.

Therefore, Applicants assert that independent claims 1, 16, and 21 contain allowable subject matter. Claims 2-15, 17-20, and 22-24 depend from independent claims 1, 16, and 21, respectively, and therefore also contain allowable subject matter. Accordingly, Applicants respectfully request that the rejections to claims 1-24 under 35 U.S.C. §103(a) be withdrawn.

With respect to independent claim 25, for similar reasons stated above, it would not be proper to combine the teaching of McDermott and Masubuchi because the references teach away from each other and should not be combined. Specifically, it would not be obvious to those of skill in the art to create a table of cache entries without addresses or cache data separate from the cache because no means for associating the table to the cache lines was taught by the prior art or known to those of ordinary skill in the art.

Therefore, Applicants assert that independent claim 25 contains allowable subject matter. Claims 26-30 depend from independent claim 25, and therefore also contains allowable subject matter. Accordingly, Applicants respectfully request that the rejections to claims 25-30 under 35 U.S.C. §103(a) be withdrawn.

For all the above reasons, the Applicant respectfully submits that this application is in condition for allowance. A Notice of Allowance is earnestly solicited.

The Examiner is invited to contact the undersigned at (408) 975-7500 to discuss any matter concerning this application.

The Office is hereby authorized to charge any additional fees or credit any overpayments under 37 C.F.R. §1.16 or §1.17 to Deposit Account No. **11-0600**.

Respectfully submitted,

KENYON & KENYON

Dated: July 26, 2004

By: _Stephen T. Neal_

Stephen T. Neal
(Reg. No. 47,815)
Attorneys for Intel Corporation

KENYON & KENYON
333 West San Carlos St., Suite 600
San Jose, CA 95110

Telephone: (408) 975-7500
Facsimile: (408) 975-7501